UNIVERSITY OF SOUTHERN DENMARK

Chapter 9
The programme specific part of the curriculum for:

MASTER OF SCIENCE (MSc) IN ENGINEERING (SOFTWARE ENGINEERING) (Civilingeniør, Cand. Polyt. i Software Engineering)

Curriculum 2015, Version 1.1

Applicable to students admitted September 2015 onwards

The curriculum is divided into general provisions (Chapters 1-8), a programme specific part (Chapter 9) and the module descriptions of the programme's individual course modules. Students should familiarise themselves with all three parts in order to get a complete overview of the provisions regulating the programme.

§1 Job Profiles

With an extensive knowledge in the field of software engineering the graduate engineer in Software Engineering is qualified for a wide variety of jobs in many different industries that use, integrate and develop software or products incorporating software. It could, for example, be in industry, finance, health, defence and security, telecommunications, transport, education, or administration.

Graduates are qualified for work in companies and organisations that develop software, including large and complex software systems. The graduate is qualified to independently take responsibility for and contribute to software development; i.e. the development of new software and customisation, integration and further development of existing software. The graduate is qualified to consider the technological, user-oriented as well as the organisational aspects of software development.

With the programme's emphasis on innovation and internationalisation, the graduate will also appeal to the many companies that develop software in a global context and use software as the basis for innovation and product development.

The graduate engineer will be qualified for a range of job functions in public and private companies, such as executive functions as software developer, system developer, system designer, system integrator, IT and system architect, product manager; organisational functions such as project manager, software quality engineer, IT strategy manager, head of project; or consulting functions such as IT consultant or system consultant. Furthermore, the graduate is qualified for jobs in research and education, for example scientific research assistant or teacher. Especially well-qualified graduates will be able to continue as PhD students in a relevant research field.

§2 Competence Profile

Purpose of the MSc in Software Engineering
The purpose of the MSc in Engineering (Software Engineering) is to, on a scientific basis, educate engineers who can independently take responsibility for and contribute to the development of new software as well as to the customisation, integration and further development of existing software in a way that considers scientific theories in the subject field, user-oriented and organisational aspects, software as an innovative element, and the impact of globalisation on software and software development (and vice versa).

The graduate must be able to independently take responsibility for and participate in the investigation of the need for the software, identification of requirements, analysis, software design, interaction design, programming and testing, as well as project management, change and configuration management, and quality management.

The graduate must be able to work systematically with large and complex software systems to control functionality (incl. security), life cycle, and software qualities important to operations (incl. user-friendliness), software development and software maintenance.

The graduate must be able to organise a well-defined process resulting in a software product with the right qualities, to the right price and at the right time. The graduate must be able to develop and deliver quality software as project manager, through individual work and as part of a team. Moreover, he or she must be able to reconcile conflicting project objectives and find acceptable compromises with limitations such as costs, time, knowledge, existing systems, organisation and environment.

Engineering competencies
The programme is structured in accordance with the educational concept ’The Engineering Education Model of the University of Southern Denmark’ or, in Danish, ’Den Syddanske Model for Ingeniøruddannelser’ (DSMI). DSMI is based on a pedagogical/didactic approach which promotes a learning environment where students during their studies acquire a variety of engineering competencies in addition to the knowledge, skills and competences related to the specific programme.

The engineering competencies that characterise the MSc in Engineering programme builds on the engineering competencies acquired on the bachelor programme.

Knowledge, skills and competencies within the scientific fields related to the programme
The competencies characterising the MSc in Software Engineering are based on and improve the skills acquired on the bachelor programme in software engineering.

MASTER OF SCIENCE IN SOFTWARE ENGINEERING	SM-ADS (1. sem)	SM-SCM (1. sem)	SM-SLC (1. sem)	SM-DSC (1. sem)	SM-PRO1 (1. sem)	SM-PRO2 (1. sem)	SM2-SWC (2. sem)	SM2-UBC (2. sem)	SM2-PRO2 (2. sem)	SM-ISS (3. sem)	SM-MT (3./4. sem)
KNOWLEDGE AND UNDERSTANDING											
possess knowledge of software lifecycle based on the highest international research, and understand and, on a scientific basis, reflect on the knowledge of software lifecycle as well as be able to identify scientific issues			X								X
possess knowledge of data science based on the highest international research, and understand and, on a scientific basis, reflect on the knowledge of data science as well as be able to identify scientific issues				X							X
possess knowledge of software ecosystems based on the highest international research, and understand and, on a scientific basis, reflect on the knowledge of software ecosystems as well as be able to identify scientific issues							X				X
possess knowledge of intelligent software based on the highest international research, and understand and, on a scientific basis, reflect on the knowledge of intelligent software as well as be able to identify scientific issues								X			X
possess knowledge of the socio-technological and socio-cultural conditions that govern the market penetration of software solutions										X	X
possess knowledge of intellectual property right (IPR), patents and software copyright, and their significance in the development of new products and services										X	X
possess knowledge of management of software as a product and has knowledge about the various business models for software delivery										X	X
possess knowledge of the specific features of software patents in emerging markets										X	X
SKILLS											
analyse and assess the technical qualities of a software-related solution	X										X
master the scientific methods and tools of software engineering and master the general skills related to employment in software engineering		X				X					X
evaluate and select among the scientific theories, methodologies, tools and general skills of software lifecycle, and set up new analysis and solution models on a scientific basis			X		X						X
evaluate and select among the scientific theories, methodologies, tools and general skills of data science, and set up new analysis and solution models on				X		X					X

MASTER OF SCIENCE IN SOFTWARE ENGINEERING	SM-ADS (1. sem)	SM-SCM (1. sem)	SM-SLC (1. sem)	SM-DSC (1. sem)	SM-PRO1 (1. sem)	SM-PRO2 (1. sem)	SM2-SWC (2. sem)	SM2-UBC (2. sem)	SM2-PRO2 (2. sem)	SM-ISS (3. sem)	SM-MT (3./4. sem)
a scientific basis											
evaluate and select among the scientific theories, methodologies, tools and general skills of software eco systems, and set up new analysis and solution models on a scientific basis							X		X		X
evaluate and select among the scientific theories, methodologies, tools and general skills of intelligent software, and set up new analysis and solution models on a scientific basis								X	X		X
communicate research-based knowledge and discuss professional and scientific issues related to software lifecycle with both peers and non-specialists			X		X						X
communicate research-based knowledge and discuss professional and scientific issues related to data science with both peers and non-specialists				X		X					X
communicate research-based knowledge and discuss professional and scientific issues related to software ecosystems with both peers and non-specialists							X		X		X
communicate research-based knowledge and discuss professional and scientific issues related to intelligent software with both peers and non-specialists								X	X		X
conduct life cycle assessments for software systems and evaluate the derived consequences for life expectancy			X		X						X
analyse conditions and obstacles to the development of innovative solutions to societal problems and effectively carry out its implementation										X	X
analyse problems of social relevance and develop software solutions that contribute to solving these problems										X	X
COMPETENCES											
master science-based methods and tools for systematic development and maintenance of large and complex software systems of an international standard	X				X						X
masters methods of survey design to evaluate the effect of software solutions in the problem domain		X				X					X
masters methods to assess the effect and value of solutions to non-trivial problems as well as compare and choose between alternative solutions		X				X					X
manage work situations and developments that are complex, unpredictable and require new solution models related to software lifecycle			X		X						X
manage work situations and developments that are complex, unpredictable				X		X					X

MASTER OF SCIENCE IN SOFTWARE ENGINEERING	SM-ADS (1. sem)	SM-SCM (1. sem)	SM-SLC (1. sem)	SM-DSC (1. sem)	SM-PRO1 (1. sem)	SM-PRO2 (1. sem)	SM2-SWC (2. sem)	SM2-UBC (2. sem)	SM2-PRO2 (2. sem)	SM-ISS (3. sem)	SM-MT (3./4. sem)
and require new solution models related to data science											
manage work situations and developments that are complex, unpredictable and require new solution models related to software ecosystems							X		X		X
manage work situations and developments that are complex, unpredictable and require new solution models related to intelligent software								X	X		X
independently initiate and carry out discipline-specific and interdisciplinary collaboration and assume professional responsibility related to software lifecycle			X		X						X
independently initiate and carry out discipline-specific and interdisciplinary collaboration and assume professional responsibility related to data science				X							X
independently initiate and carry out discipline-specific and interdisciplinary collaboration and assume professional responsibility related to software eco-systems							X		X		X
independently initiate and carry out discipline-specific and interdisciplinary collaboration and assume professional responsibility related to intelligent software								X	X		X
independently take responsibility for own professional development and specialization related to software lifecycle			X		X						X
independently take responsibility for own professional development and specialization related to data science				X		X					
independently take responsibility for own professional development and specialization related to software ecosystems							X		X		
independently take responsibility for own professional development and specialization related to intelligent software								X	X		
facilitate innovation processes and assess the level of innovation in products and services through novelty searches in patent and research databases in the relevant subject fields										X	
masters analytical methods to examine and uncover socio-technological con-texts in socially relevant issues where interaction between technology and society are deciding factors in the development of usable solutions										X	
manage and participate in global and multicultural development projects										X	
manage and perform an innovation and development process that results in a										X	

MASTER OF SCIENCE IN SOFTWARE ENGINEERING	SM-ADS (1. sem)	SM-SCM (1. sem)	SM-SLC (1. sem)	SM-DSC (1. sem)	SM-PRO1 (1. sem)	SM-PRO2 (1. sem)	SM2-SWC (2. sem)	SM2-UBC (2. sem)	SM2-PRO2 (2. sem)	SM-ISS (3. sem)	SM-MT (3./4. sem)
novel product or service that addresses an existing or emerging market.											
plan major development projects in collaboration with other professions											X
acquire research results and translate them into new products and services											X

§3 The Subjects of the Study Programme

The study programme has five subject areas:

- Software Lifecycle
- Mobile and Ubiquitous Computing
- Software Ecosystems
- Data Science
- Innovative Software Solutions

Software Lifecycle
Software products are embedded into a comprehensive life cycle, which comprises pre-development, development and post-development processes as well as project management and integral processes. Each phase of the software life cycle consists of manifold processes like concept exploration, requirements, design, implementation, verification, validation, operation and evolution. Each phase consumes and produces artefacts that are subject to these processes.

The Software life cycle subject area aims to study the various aspects of the software life cycle, and to understand and improve the respective activities, and related challenges, methods, technologies, and tools in order to create complex software systems and pave the way for optimization and innovation.

Mobile and Ubiquitous Computing

Mobile and ubiquitous computing consider how computing evolves into the fabric of everyday life by being mobile for people on the move or being embedded in their everyday surroundings. Therefore software has to be designed for and consider the interactions and continuous changes in the physical world and adapt to user intentions and context. Furthermore, software has to manage the availability, operation and reliability of resources including processing units, battery power, and user interaction, communication, sensing and actuation technologies. Key challenges include the development of flexible, scalable and decentralized applications embedded in a heterogeneous environment of resources, services and systems, e.g., in an Internet of things.

Software Ecosystems
Software exists in an ecosystem of applications, systems and resources. Successful software has to be designed for and consider the interactions and continuous changes in ecosystems of software. Software has to manage the availability and security of resources including processing units, user interaction, communication, sensor and actuation technologies. A key challenge is the development of flexible, scalable and decentralized applications embedded in software ecosystems.

Data Science
Data science is the extraction of knowledge from data. In particular it focuses on the collection, filtering, processing, creation and distribution of data. Dramatic growth in the scale and complexity of data that can be collected and analyzed is affecting all aspects of work and society. This implies that development of effective and ethical ways of using vast amounts of data is a significant chal-

lenge to science and to society as a whole. Therefore, the key challenge of data science is developing scalable techniques for data analysis and decision making, which requires interdisciplinary research in many areas, including machine learning, algorithms, statistics, operations research, databases, complexity analysis, visualization, and privacy and security.

Innovative Software Solutions

Software enables the creation of new and unforeseen solutions to automation of activities and processes that was not possible with mechanical or analogue technologies. It is playing an increasing role in realizing innovation, and also requires a radical shift in the mindsets of people to understand and explore the endless possibilities. New innovative software solution is demanded to addresses the societal challenges of the 21st century for an existing or emerging market.

§4 Programme Structure

The programme consists of three parts:

- Compulsory part
- Elective part
- Master Thesis

The compulsory part consists of compulsory modules, which constitute the programme and its special professional skills and identity within the field of software engineering. The compulsory modules are introductory, core or advanced.

The elective part consists of modules, which students are free to choose.

The Master thesis synthesizes the students' skills in a specialized contemplation of a particular theme within software engineering.

<table>
<tr><td>4th semester</td><td colspan="2">Master Thesis*</td></tr>
<tr><td>3rd semester</td><td>Compulsory Part, Advanced</td><td rowspan="2">Elective</td></tr>
<tr><td>2nd semester</td><td rowspan="2">Compulsory Part, Core</td></tr>
<tr><td>1st semester</td><td>Compulsory Part, Introductory</td></tr>
</table>

§5 Programme Structure and Modules

<table>
<tr><td>SEMESTER</td><td colspan="30">MODULES</td></tr>
<tr><td>4th semester</td><td colspan="30">Master Thesis*</td></tr>
<tr><td>3rd semester**</td><td colspan="15">SM-ISS
Innovative Software Solutions</td><td colspan="5">Elective/
In-company period*</td><td colspan="5">Elective/
In-company period*/
Master Thesis</td><td colspan="5">Elective/
In-company period*/
Master Thesis</td></tr>
<tr><td>2nd semester</td><td colspan="5">SM2-SWC
Software
Customization</td><td colspan="5">SM2-UBC
Ubiquitous
Computing</td><td colspan="5">SM2-PRO
Project in
SWC and UBC</td><td colspan="5">Elective</td><td colspan="5">Elective</td><td colspan="5">Elective</td></tr>
<tr><td>1st semester</td><td colspan="5">SM-SCM
Scientific Methods</td><td colspan="5">SM-DSC
Data Science</td><td colspan="5">SM-PRO2
Project in
SCM and DSC</td><td colspan="5">SM-PRO1
Project in
ADS and SLC</td><td colspan="5">SM-SLC
Software Lifecycle</td><td colspan="5">SM-ADS
Advanced Software
System Design and
Technologies</td></tr>
<tr><td>ECTS</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td></tr>
</table>

*) If the thesis is of an experimental nature, the student may choose to use the elective 10 ECTS on the 3rd semester as part of the thesis. This will extend the scope of the thesis to 40 ECTS. Students on a 4+4 PhD programme may use their 15 ECTS electives on third semester together with the 30 ECTS on fourth semester on a 45 ECTS master thesis. If the student selects the In-company Period, the student is not able to do a 40 ECTS thesis, because the entire 15 ECTS electives on the third semester are spent on the In-company Period.

**) Alternatively, choose exchange period at a foreign university.

§6 Semester Description

Compulsory part

Introductory modules: the student's knowledge of and practice in software engineering and project management is consolidated in the module "Advanced Software System Design and Technologies", which utilizes the prerequisites for the master programme. As the basis for further studies, the student is introduced to scientific methods in another introductory module. The introductory modules are in the 1st semester.

Core modules: Through the core modules Software Lifecycle, Ubiquitous Computing, Software Customization, Scientific Methods and Data science the student dives into selected subject areas in core software engineering research fields. The student also has the opportunity to gain deeper knowledge, skills and competencies through projects spanning subject areas. The core modules are in the 1st and the 2nd semester.

Advanced modules: Through the advanced module the student gains a complete collection of techniques for incremental and iterative development of large software projects and gets the competencies to perform an innovation and development process that results in a novel product or service that address an existing or emerging market. The advanced modules are in the 3rd semester.

Elective part

The elective part consists of modules, which students are free to choose. The elective modules are In-Company Period module and a pool of elective modules. The In-Company Period module enables the students to work part-time for a company as part of their 3rd semester.

In-Company Period

The In-Company period provides students with knowledge and understanding of practical situations, methods, processes and engineering functions of a company and enables them to apply their academic knowledge as well as prepare them for a future employment as graduate engineers.

The Master Thesis

The Master Thesis demonstrates students' complete engineering skills in a relevant, limited, engineering subject within software engineering

§7 Qualifying Degrees for Admission

7.1 Qualifying degrees

Based on 7.2 – 7.4 the university has assessed that the below degrees qualify for admission to Master of Science in Engineering (Software Engineering). The list is not exhaustive.

- BSc in Engineering (Software Engineering) – University of Southern Denmark (legal entitlement for admission)

7.2 Level and content of qualifying degrees

Qualifying bachelor and professional bachelor degrees in the scientific and technical area where the level and content of the scientific and technical courses correspond to a bachelor of science degree or a bachelor of engineering degree in the subject area of the MSc in Engineering (Software Engineering) programme.

7.3 Academic content of qualifying degree

MSc in Engineering (Software engineering) admits applicants with a bachelor degree or a professional bachelor degree in the subject area of the program cf. 7.2 provided that the degree covers:

Subject knowledge	**Extent**
Software Development and Programming	20 ECTS
Operating Systems	5 ECTS
Distributed Systems and Networking	5 ECTS
Database Systems	5 ECTS
Algorithms and Data Structures	5 ECTS
Artificial Intelligence	5 ECTS

7.4 Additional courses

Should the applicant's degree fail to meet the requirements mentioned in 7.1 - 7.3, it is possible to acquire the necessary skills through additional courses offered at the University of Southern Denmark. The extent of additional courses cannot exceed 15 ECTS.

Additional courses have to be taken after admission to the programme. The courses can be taken during the first two semesters of the programme and must be passed by the end of the first year of study. Additional courses are restricted to courses offered by the University of Southern Denmark as summer courses or parallel to the first year of the master programme.

7.5 Admission with a foreign degree

Applicants with a bachelor degree or professional bachelor degree from a foreign university who meet the requirements of 7.2 and 7.3 are eligible for admission subject to an academic assessment and comparison of whether the applicant's academic qualifications correspond to those of qualifying Danish degree.

7.6 Possible exemptions

Applicants whose bachelor degree or professional bachelor degree fails to meet the terms stated in 7.1 – 7.5 are not eligible for admission.

Applicants who do not hold a bachelor degree or a professional bachelor degree but who have the academic qualifications equivalent thereto are eligible for admission should their qualifications, based on an academic assessment and comparison, correspond to those of a qualifying Danish degree.

Two-year transitional arrangement regarding additional courses:
Completed and passed additional courses, i.e. single courses from existing bachelor programmes, may be included in the application for admission until 31 August 2016.

§8 External Examiners and Study Board

The study programme belongs under the Academic Study Board of the Faculty of Engineering and the Danish corps of external examiners for engineering programmes. Modules offered by the Faculty of Science belong under the corps of external examiners for science.

§9 Entry into Force

1. Approved by the Academic Study Board of the Faculty of Engineering and the Director of Studies on behalf of the Dean of the Faculty of Engineering on 18 April 2013.
2. Curriculum 2014 approved by the Academic Study Board of the Faculty of Engineering and the Director of Studies on behalf of the Dean of the Faculty of Engineering on 23 June 2014 (Version 1.0)
3. Curriculum 2015 approved by the Academic Study Board of the Faculty of Engineering and the Director of Studies on behalf of the Dean of the Faculty of Engineering on 18 March 2015 (Version 1.0).
4. Amendments approved by the Academic Study Board of the Faculty of Engineering and the Director of Studies on behalf of the Dean of the Faculty of Engineering on 20 October 2015 (Version 1.1).